REMARKS

This Amendment is responsive to the Office Action dated February 21, 2006. Applicant has amended claims 1, 9, 13, 17, 18, 22, 24, 28 and 35,and cancelled claims 21 and 27. Claims 1-9, 11-20, 22-26, 28-32 and 34-35 are pending.

Claim Rejection Under 35 U.S.C. § 103

In the Final Office Action, the Examiner rejected claims 1-9, 11-32, 34-35 under 35 U.S.C. 103(a) as being unpatentable over Mathur (USPN 6,424,658) in view of Muller et al. (USPN 6,245,680). Applicant respectfully traverses the rejection to the extent applicable to claims as amended.

In the Office Action, the Examiner argued that Mathur discloses a routing component having substantially all of the features of Applicant's independent claims, but acknowledged that Mathur utilizes only internal memory and does not disclose use of an external memory. With respect to these features, the Examiner cited Muller as disclosing a routing element that utilizes external memory, and concluded that it would have been obvious to one of ordinary skill to modify the Mathur routing component in view of Muller to achieve Applicant's claimed invention.[1]

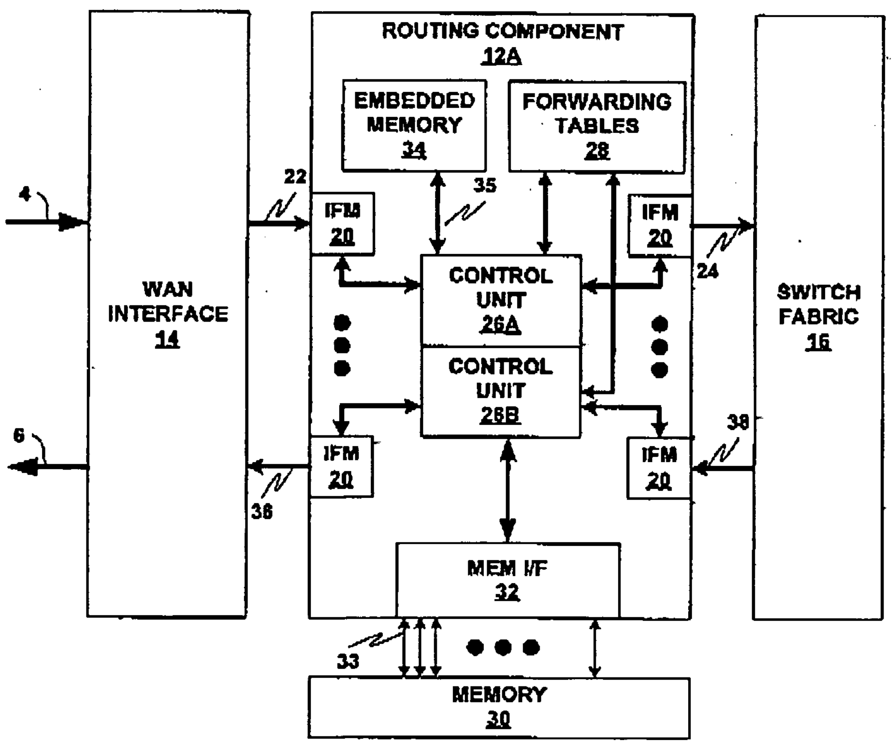
In forming the rejection, the Examiner reasoned that, in Mathur, the interface by which a packet is received determines the direction of communication.[2] Thus, the Examiner's argument is essentially that if one of ordinary skill were to modify Mathur in view of Muller to utilize both embedded and external memory, then the resultant routing component would inherently include a controller that stores packets differently (i.e., in external or internal memory) based on the direction because the interface on which the packet was received determines the direction.

To further clarify the differences between Applicant's claimed invention and the prior art, Applicant has amended the independent claims 1, 9, 18 and 24 to include elements similar to currently pending independent claim 30. Claim 30 is directed to a method that requires accessing a forwarding table with a control unit of the network router to determine a network <u>destination</u> for the data. Claim 30 further requires, when the <u>destination</u> requires forwarding the data to a

[1] See Office Action pg. 3.
[2] Id at pg. 3, paragraph 2.

second routing component internal to the router using a switch, buffering the inbound data within an embedded memory internal to the first routing component. In addition, amended claim 30 requires, when the destination requires forwarding the outbound data to the network interface, buffering the outbound data within a memory external to the first routing component.

In other words, claim 30 is directed to an embodiment in which data (e.g., packets) received on the same interface may be buffered differently, i.e. using either internal or external memory, based on the particular destination of the packet. To illustrate this by way of example in reference to the present application, Applicant directs the Examiner to FIG. 2, reproduced below:



The present application discloses that routers forward packets based on destination information within the packets in accordance with forwarding tables.[3] With respect to FIG. 2, the present application states that upon receiving an inbound packet via input link 4, routing component 12A may forward the inbound packet directly to outbound link 6 via WAN interface 14 or to another routing component via internal switch fabric 16.[4] An example embodiment is described in

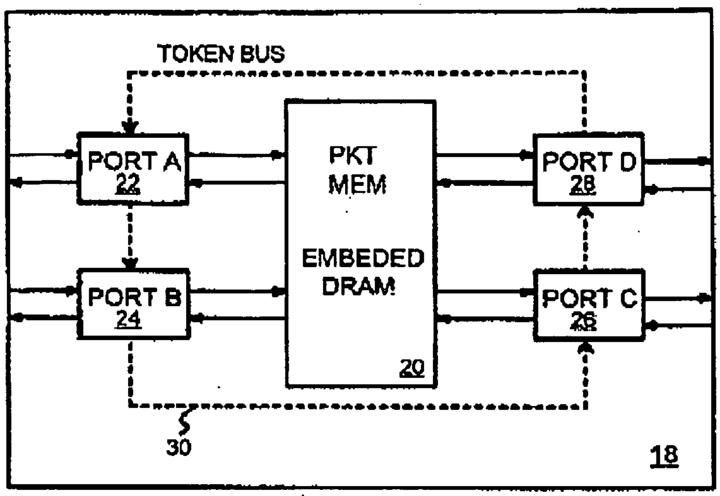
[3] Pg. 1, ll. 15-20.
[4] Pg. 6, ln. 26-pg. 7, ln. 5.

which the functionality of control units may include a packet forwarding engine that either directs inbound packets to outbound link 6 <u>or</u> to other routing components via switch fabric 16 based on forwarding tables 28. Thus, in such an embodiment, data received on the <u>same</u> interface (e.g., inbound packets received from link 4) may be forwarded directly back to the network as outbound packets using a network interface of the same bandwidth (e.g., WAN interface 14) <u>or</u> forwarded to another routing component by the internal switch fabric 16 having a different bandwidth. In this regard, routing component 12A actively determines the destination for the particular data (e.g., packet) and forwards the data accordingly. As described at length in the application, the data is buffered differently (i.e. using either internal or external memory) based on the particular forwarding direction and the bandwidth differences between the interfaces.

Mathur in view of Muller fails to teach or suggest a routing component in which data (e.g., packets) received on the <u>same</u> interface may be buffered differently, i.e. using either internal or external memory, based on the particular destination of the packet. That is, with respect to claim 1 for example, Mathur in view of Muller fails to teach or suggest a routing component of a router having at least one control unit that buffers data received from a network using embedded memory internal to the integrated circuit when the destination requires forwarding the data to a second routing component of the router using a switch, and buffers the data in the external memory when the destination requires forwarding the data to the network via the first interface. For these or similar reasons, Mathur in view of Muller fails to teach or suggest the elements of independent claims 9, 18, 24, 30 and 35.

Mathur (USPN 6,424,658)

Mathur describes a store-and-forward network switch that uses an embedded dynamic-random-access memory (DRAM) packet memory. In particular, FIG. 2 of Mathur shows a network switch chip 18 that receives packets from one of four ports A, B, C, D and stores the packets in embedded DRAM packet memory 20. The network switch chip 18 transmits the stored packets out to one or more of the four ports A, B, C, D. The following illustrates FIG. 2 of Mathur:



Mathur makes clear that <u>all</u> packets forwarded between ports 22-28 are stored in embedded packet memory 20. For example, at col. 6, ll. 3-7, Mathur specifically states the following:

> *Port logic 22, 24, 26, 28 are bi-directional ports to a network node connected to a computer, peripheral, LAN segment, or other network equipment such as another switch, router, repeater, bridge or hub. Packets may be input or output from any port. When a packet is received by port logic 22, 24, 26, 28, it first writes the packet into embedded DRAM packet memory 20.*

Thus, as stated above and illustrated in FIG. 2 (above), Mathur teaches a switch in which packets forwarded between any of the four interface ports 22-26 are buffered within an embedded packet memory 20 regardless of the forwarding operation, and no controller is used to buffer packets from the same interface differently based on any criteria whatsoever.

Muller et al. (USPN 6,246,680)

Muller describes a highly integrated multi-layer switch element. According to Muller, the switch element includes multiple ports for transmitting and receiving packets over a network. FIG. 2 of Muller illustrates the described switch element:

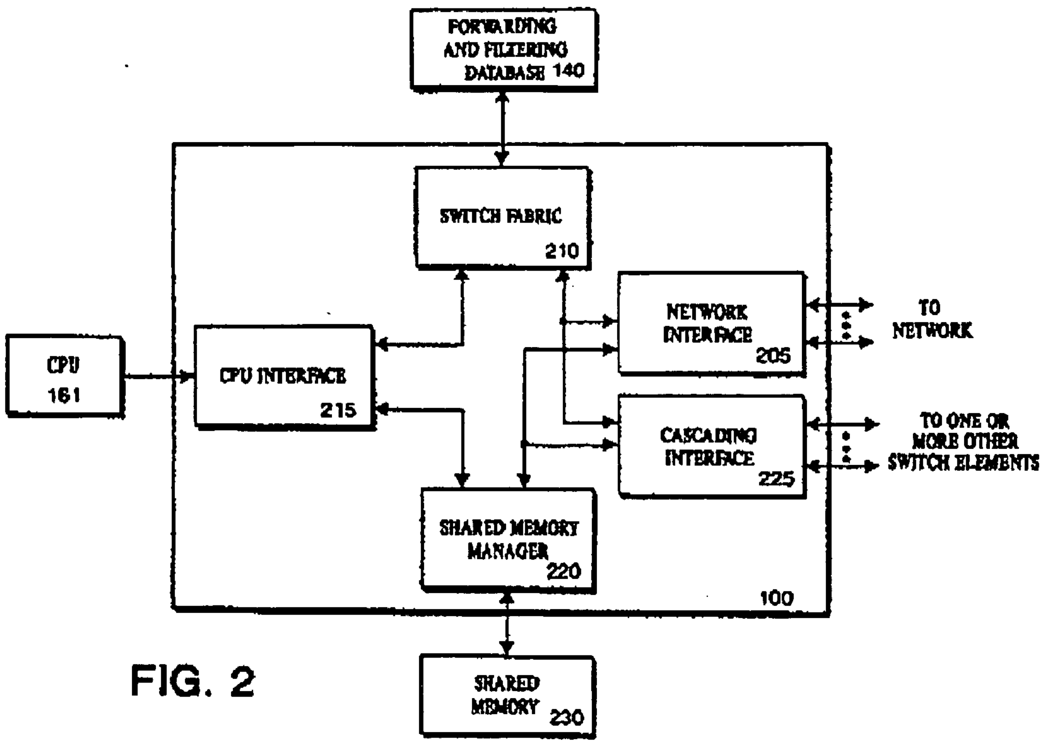


FIG. 2

Unlike Mathur, Muller makes use of an external shared memory 230 for buffering all packets flowing between any of the network interfaces 205. For example, col. 1, ll. 41-60 of Muller states:

> *Input packet processing includes the following: (1) receiving and verifying incoming Ethernet packets, (2) modifying packet headers when appropriate, (3) requesting buffer pointers from the shared memory manager 220 for storage of incoming packets, (4) requesting forwarding decisions from the switch fabric block 210, (5)* ***transferring the incoming packet data to the shared memory manager 220 for temporary storage in an external shared memory 230****, and (5) upon receipt of a forwarding decision, forwarding the buffer pointer(s) to the output port(s) indicated by the forwarding decision. Output packet processing may be performed by one or more output ports of the network interface 205. Output processing includes* ***requesting packet data from the shared memory manager 220****, transmitting packets onto the network, and requesting deallocation of buffer(s) after packets have been transmitted.*

Thus, Muller teaches a switch in which an external shared memory 230 is used to buffer all packets flowing in between network interfaces 205, and no controller is used to buffer packets from the same interface differently based on any criteria whatsoever.

As a result, Mathur in view of Muller fails to teach or suggest a routing component in which data (e.g., packets) received on the same interface may be buffered differently, i.e. using either internal or external memory, based on the particular destination of the packet. For these reasons, neither Mathur nor Muller, either singularly or in combination, teach or suggest a routing component having a controller that buffers the data from a first interface using the embedded memory internal to the routing component when a destination of the data requires forwarding the data to a second one of the routing components using the crossbar arrangement, and buffers the data in the external memory when the destination requires forwarding the data to the network via the first interface, as required by amended claim 1.

Applicant submits that the basis for the rejection is now moot. As noted above, the rejection is hinged on the premise that modification of Mathur in view of Muller would result in a routing component that stores packets differently because, inherently, the interface on which the packet was received determines the direction. Neither Mathur nor Muller describes any a routing component in which data (e.g., packets) received on the same interface may be buffered differently, i.e. using either internal or external memory, based on the particular destination of the packet. Modification of Mathur to incorporate external memory and "per-port buffering," as suggested by the Examiner, fails to achieve a routing component that actively selects different memory buffering schemes for data received from the same interface.

CONCLUSION

All claims in this application are in condition for allowance. Applicant respectfully requests reconsideration and prompt allowance of all pending claims. Please charge any additional fees or credit any overpayment to deposit account number 50-1778. The Examiner is invited to telephone the below-signed attorney to discuss this application.

Date: May 18 2006

By: Kent J. Sieffert

SHUMAKER & SIEFFERT, P.A.
8425 Seasons Parkway, Suite 105
St. Paul, Minnesota 55125
Telephone: 651.735.1100
Facsimile: 651.735.1102

Name: Kent J. Sieffert
Reg. No.: 41,312